SMART REACH: OFFLINE SMS-BASED REMOTE ACCESS SYSTEM FOR ANDROID DEVICES

[1]M.SION KUMARI , [2]MOHITHA BATTU , [3]GAYATHRI BONELA , [4]BORAPUREDDY TEJASHWINI , [5]PRATHYUSHA BOYINA

[1]Assistant. Professor, Department of Computer Science and Engineering
[2345] Students, Department of Information Technology and Computer Applications ,
Andhra university college of engineering for women ,Visakhapatnam - 530003, AP India
[5] boyinaprathyusha0506@gmail.com

Abstract*—This paper is about a system that lets you control your Android phone from another phone using text messages. The system is helpful when you cannot use your phone directly. For example when you forget your phone at home or lose it you can use this system to do things on your phone. You can send text messages from any phone to make your other phone do things. These things include getting the phone numbers of your friends finding out where your phone is switching the sound on seeing how much battery is left getting help and locking the screen so nobody else can use your phone. Each time you send a text message to control your phone the system checks a code to make sure it is really you. This system is different from systems because it does not need the internet to work. It uses the network that your phone uses to make calls. This means you can use the system even when you do not have internet. The people who made this system used a tool called Android Studio and two programming languages, Java and XML. The system is good because it is safe reliable and does not cost a lot of money. It is a way to control your Android phone from another phone. The system is an Android mobile access system that uses Short Message Service as a way to communicate. The remote Android mobile access system is a cost-solution, for remote mobile device management. The remote Android mobile access system provides an secure way to control your Android phone.

***Keywords*—Android Application, Android Studio, GSM Network, Java, Mobile Device Security, Remote Access, SMS-based Communication, XML**

I. INTRODUCTION

Smartphones are a part of our daily lives now. We use them to talk to people store information and manage our tasks. People keep a lot of things on their phones like contacts, messages and personal documents.. Sometimes we lose our phones at home at work or in public places. This makes it hard for us to get to our information or use our phone.

When we lose our phone it might be on mode, which makes it hard to find.. Someone else might find it and look at our private stuff. This is not good for our privacy and security.

Most of the time we need the internet to access our phone from another device.. The internet is not always available, like in emergencies or places with bad internet. Other ways of connecting devices, like Bluetooth and Wi-Fi do not work well over distances. We need a way to access our phone that does not need the internet all the time.

This is why we made a system that lets us access our Android phone from another device using SMS. We can send messages to our phone to do things like get our contacts read messages see where our phone is and change settings. Only people we allow can send these messages so our information is safe.

Our system works whether we have the internet or not. It is a cheap and reliable way to manage our phone from another device. Because it uses the network our phone uses to make calls we can use it anywhere. This makes it easier for us to manage our phone especially if we lose it. It also helps keep our phone and information safe.

II. LITERATURE REVIEW

Recent studies in mobile device management have looked at systems that work without internet connection. Joy et al. [1] Suggested a system that lets users get device information using SMS commands. Tupe et al. [2] Also made a system for Android phones that uses SMS for access with secure logins. Their system focuses on being reliable and easy to use in areas with network. Earlier studies by Mahadule et al. [3] And Joshi et al. [4] Showed that SMS can be used to control smartphones. They

found that users can access contacts get device info and control functions using SMS commands. Jain and Shanbhag [5] used SMS for home automation and monitoring. Kumar et al. [6] Also used SMS for IoT systems. In both cases SMS worked well when internet was not available.

Some researchers like Khan et al. [7] Have used SMS to track locations. They proved that SMS can be a backup when GPS data can't be sent due to lack of internet. The Android developer guides [8] , [9] also provide info, on how to implement permissions and SMS handling. Many current systems still need internet to work or don't have strong security. Their user interfaces are also not very user-friendly. Can't do many things. These limitations show that we need a system that's secure, efficient and works fully offline. This is what motivated us to develop our solution.

Despite these advancements, many existing systems either partially depend on internet connectivity or lack comprehensive security features such as strong authentication and controlled access mechanisms. Furthermore, user interface design and integration of multiple functionalities remain limited in several implementations. These limitations highlight the need for a more secure, efficient, and fully offline-capable system, which motivates the development of the proposed solution.

III. METHODOLOGY

The Smart Reach system is made to give people control over their Android smartphones from away using text messages. The main idea of Smart Reach is to let people use their phone even when they cannot find it. It is not near them. Smart Reach uses text messages to talk to the phone, which makes it work better and is available all the time.

The Smart Reach app will be put on the Android phone. Will work in the background without getting in the way of normal phone use. The app will look at every text message that comes in. When it gets a message it will check if the message looks like a command. To keep things Smart Reach will check who sent the message using a secret code or a trusted phone number before doing what the message says.

After Smart Reach checks who sent the message it will do what the person asked for. The person can lock the phone change how it works find where the phone is and look at phone numbers. Smart Reach uses tools on the Android phone to do these things. These tools include a message sender, a location finder and a security tool to lock the phone. After Smart Reach does what the person asked for it will send a message to say it is done.

One good thing about Smart Reach is that it does not need the internet to work. It only uses text messages that come over the phone network. This makes Smart Reach work better in places where the internet's not available. Smart Reach is also easy to use because it has simple text message commands that people can remember so they can use it without any problems.

Overall Smart Reach is made to be a way to control smartphones from away that people can trust and use easily. With text messages security checks and Android tools Smart Reach is a way to manage smartphones from away, in real life using the Smart Reach system and the Smart Reach application.

IV. System Architecture And Design

a) System Architecture

Fig. 1. The Smart Reach system is based on a layered architecture, and its application is divided into different levels. Smart Reach has four layers, namely, the User Interface Layer, SMS Command Processing Layer, Service Layer, and Device Functions & APIs Layer. Each layer has a specific role to perform.

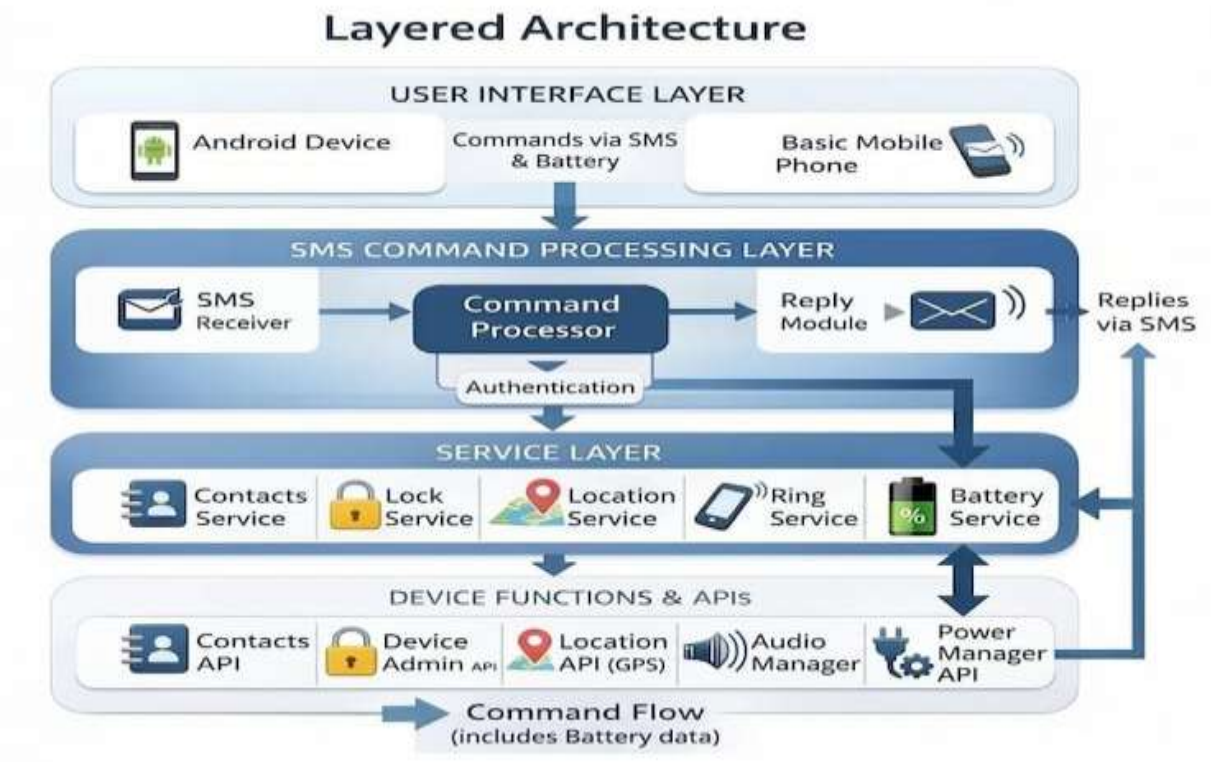

Fig 1. Layered architecture of the Smart Reach system.

The User Interface Layer is the beginning of the system. In this layer, a legitimate user sends a command from a mobile device through SMS. The command message contains the user's passcode and a predefined word such as location, contact, lock, or ring. This message is transmitted through the GSM network to the Android device on which the Smart Reach application is installed.

The SMS Command Processing Layer is responsible for handling the messages. The SMS Receiver uses the Android Broadcast Receiver to detect the messages. The Command Processor reads the message content and checks the passcode using an authentication method. The Command Processor also identifies the action requested. Finally, the Reply Module creates a reply message that will be sent back to the user.

The Service Layer carries out the requested actions, which include accessing contacts, locking the phone, tracing the location, and controlling the ringtone.

Lastly, the Device Functions and APIs Layer directly interacts with Android system APIs like the Contacts API, Device Admin API, Location API, and Audio Manager to perform actual device operations.

B. System Components

The Smart Reach system has four major software modules that work in concert to process SMS commands and perform remote device operations.

- SMS Receiver Module:

This module utilizes an Android BroadcastReceiver to recognize and intercept SMS messages. Once an SMS is received, it is forwarded to the command parsing component for further processing.

- Command Parser and Authenticator:

This module parses the passcode and command identifier from the SMS message received. It authenticates the passcode with the stored security credential and identifies the type of command being sent by the user.

- Command Executor:

Once authentication is successful, this module communicates with Android system APIs to execute the command sent by the user. The command may include retrieving the location of the device, accessing stored contacts, setting up audio settings, or locking the screen of the device.

- Response Dispatcher:

Once the command is executed successfully, this module sends a response SMS to the originating phone number with the result of the executed command.

Together, these modules allow the Smart Reach system to securely receive commands, process them efficiently, and provide responses to the authorized user.

C. Comand Protocol

The Smart Reach system has a defined format for SMS commands that ensure proper communication between the authorized user and the target device. The format of a command message is composed of two major parts: a passcode and a command identifier. The passcode is for authentication, while the command identifier is for defining a specific action to be performed by the target device.

The command message format is:

<passcode> <command-id>

In this format the passcode is the security code that the user entered at the time of configuring the application for use. The command-id is a keyword that has been defined previously and will instruct the system to carry out a function. When the Smart Reach application receives an SMS, it extracts both the passcode and the command ID from the message, checks the validity of the passcode, and carries out the function specified in the command ID if the authentication is successful.

This ensures that communication between the authorized user and the target device is secure and organized. The command protocol prevents unauthorized users from sending commands to the device through the use of a passcode and command identifiers. Each command keyword is associated with a unique function that is available within the Smart Reach application. Once a command is sent to the system and validated, the system processes the request and activates a service module to perform the required function.

A table (Table I) is provided that shows the supported SMS commands and associated system functions.

TABLE I. SUPPORTED SMS COMMANDS

Command ID	**Operation**	**Example SMS**	**Response**
LOCK	LOCK SCREEN	1234 LOCK	Device locked successfully
RING	ENABLE RING MODE	1234 RING	Phone ringing activated
LOCATION	RETRIEVE LOCATION	1234 LOCATION	Latitude and Longitude sent via sms
CONTACT	RETRIEVE CONTACTS	1234 CONTACT JOHN	Contact number with phone number
BATTERY	BATTERY STATUS	1234 BATTERY	Shows battery percentage
SMARTREACH HELP	SHOW AVAILABLE COMMANDS	1234 SMARTREACH HELP	List of all supported commands via SMS

D. Sms Communication Flow

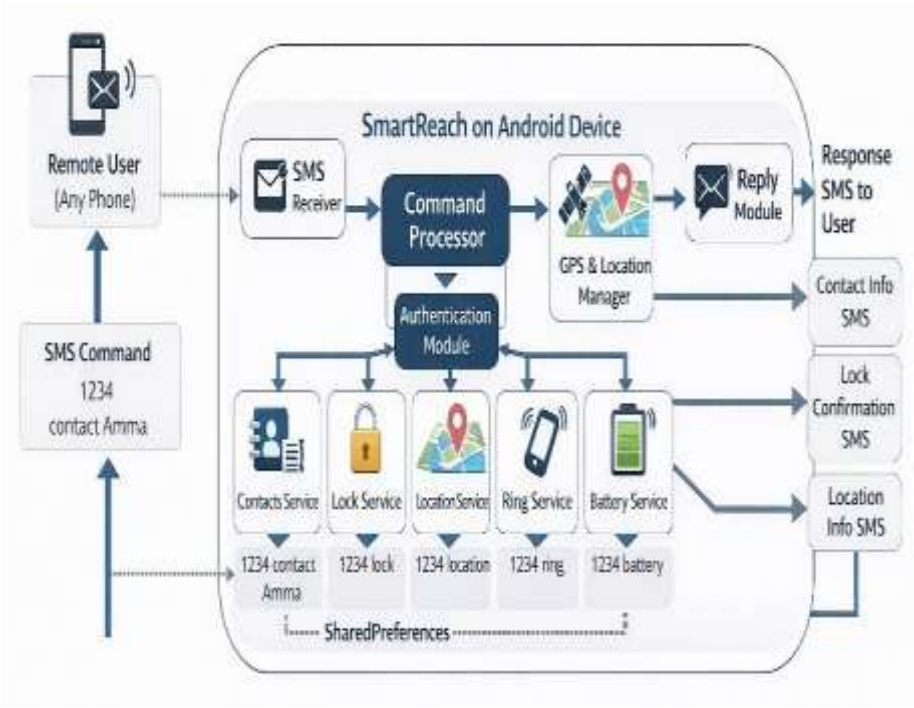

Fig 2. SMS communication flow of the Smart Reach remote access system.

The SMS communication flow demonstrates the process of how the command is sent from the authorized user to the Smart Reach application and how the process works in executing the operations on the target device. The process starts with the authorized user sending a command through SMS from any mobile device. The SMS contains a passcode followed by a command word such as contact, location, lock, or ring. The SMS is then sent to the Android device through the GSM network.

Once the SMS reaches the device, the Smart Reach application captures the SMS using the SMS Receiver module. The SMS is then sent to the authentication module, where the passcode in the SMS command is verified.

Once the authentication process is successful, the command processor processes the command keyword and determines the action required. Depending on the command, the system calls the service module to execute the requested action. These service modules may include retrieving contact numbers, locking the device, accessing the GPS service to get the device's location, or activating the ring service.

Once the action is complete, the reply module sends a reply message through SMS to the user. This process enables the Smart Reach system to function efficiently without the need for an internet connection.

V. IMPLEMENTATION

A. Android

Smart Reach is an Android-based application. The Android application is developed using Android Studio. Android has a rich, versatile, flexible, and highly accessible API that allows you to build applications that can interact with mobile device hardware and services of the system.

In this project, a few important features are implemented using Android APIs like SMS Monitoring, location info, device lock, contacts. Android allows running background parts of an application like Broadcast Receiver to listen to receive an SMS even if the application is not running.

The application is also utilizing Android activity structures that allow us to differentiate between different screens like login, registration, dashboard, instruction pages. User-friendly activities to configure your applications well these days are over.

Another important part of this system is that it is utilizing “Device Policy Manager” of Android. The application is able to lock the device remotely after receiving a valid command. The Android permission protocol is that it needs to ask users for permission before it is able to use any private resources like contacts, SMS messages, or location.

The advantages of using Android Platform are that it is a small application that is able to run on a variety of Android phones.

B. Java

The programming language Java is utilized for the development of the Smart Reach application. Java is a programming language that offers various features of object-oriented programming. These features are very useful for managing the complexities of Application Logic.

In this project, the fundamental system parts like SMS processing, authentication, parsing commands, and performing device command operations are developed using Java. Java is utilized for capturing SMS messages in the background and handling them immediately using Broadcast Receiver written in Java.

The Java code is also utilized for checking the passcode stored as shared preferences by the application. Once a message is received, it verifies the passcode from the SMS and then verifies if it is a valid command.

Various Java classes are incorporated into the system for handling different functionalities of the system. In this regard, it is observed that the location module is utilizing GPS coordinates from Android location services, while the contact module is utilizing the device's contact database for phone numbers.

C. Xml

XML is applied in Smart Reach app to lay out the UI of the system. In Android, XML layouts are used to define the appearance of the UI elements on the screen and how they are placed relative to each other in the app.

This project uses XML to make layouts for multiple screens like login page, registration page, dashboard, instruction screen and reset password screen. UI elements like text fields, buttons, images and card views are common to all the layouts.

For instance, the dashboard layout has card views for common application functionalities like contacts, location, phone lock, sound profile, etc. When the user clicks these cards, the app displays instructions for sending the right SMS command.

Also, XML allows a clear separation between the UI itself and the logic of the application which is executed by Java. This division helps in maintaining and modifying the application.

Moreover, XML enables developers to set styles, colors, and dimensions to add more visual appeal to the application. Using XML layouts, the Smart Reach application achieves this with a clean and friendly-looking interface that makes it easy for the user to set up and see the system.

D. Permissions

Permissions play an important role in ensuring that the Smart Reach application can safely access the resources required to perform its remote monitoring functions. Android follows a permission-based security model in which applications must request approval from the user before accessing sensitive device data or system features. This mechanism helps protect user privacy while allowing applications to perform specific operations necessary for their functionality.

The Smart Reach application requires several permissions in order to implement its SMS-based remote control features. One of the most important permissions is related to SMS communication. The application needs permission to receive SMS messages so that it can detect incoming commands sent by the authorized user. It also requires permission to send SMS messages in order to deliver response messages that contain requested information such as location links or confirmation messages.

Location permissions are required to access the device's GPS or network location services. These permissions allow the application to determine the current position of the device and generate a Google Maps link that can be sent back to the user through SMS. Contact access permission is also necessary because the system retrieves phone numbers from the device's contact list when a contact command is received.

Another essential permission is device administration permission, which enables the application to lock the device remotely through the Android Device Policy Manager. Together, these permissions allow the Smart Reach system to perform its operations securely while maintaining proper control over device resources.

E. Sms Manager

The SMS Manager API is essential to the Smart Reach system. It lets the application send SMS messages automatically. This feature allows the device to respond right away when it receives a command from an authorized user.

When the application processes an SMS command correctly, it creates a response message. This message either contains the requested information or confirms the action taken. The SMS Manager API sends this message back to the sender's phone number.

For instance, if a user sends a command asking for the device's location, the application gets the GPS coordinates and turns them into a Google Maps link. The SMS Manager then sends this link back to the user, making it easy for them to see the device's location.

Likewise, when the contact command is used, the application looks up the phone's contact database and sends the matching contact number via SMS. If the user sends a lock command, the system locks the device and sends a confirmation message.

The SMS Manager API ensures dependable communication between the authorized user and the device. Since SMS works over the GSM network, the system stays functional even without internet connectivity.

VI. Results And Discussion

The smart reach application starts with a secure login interface where the user is required to provide valid email and password credentials. This ensures that only authorized users have access to the application and can remotely control the device. As shown in Fig. 3, the interface provides a simple and secure method of inputting user credentials for verification.

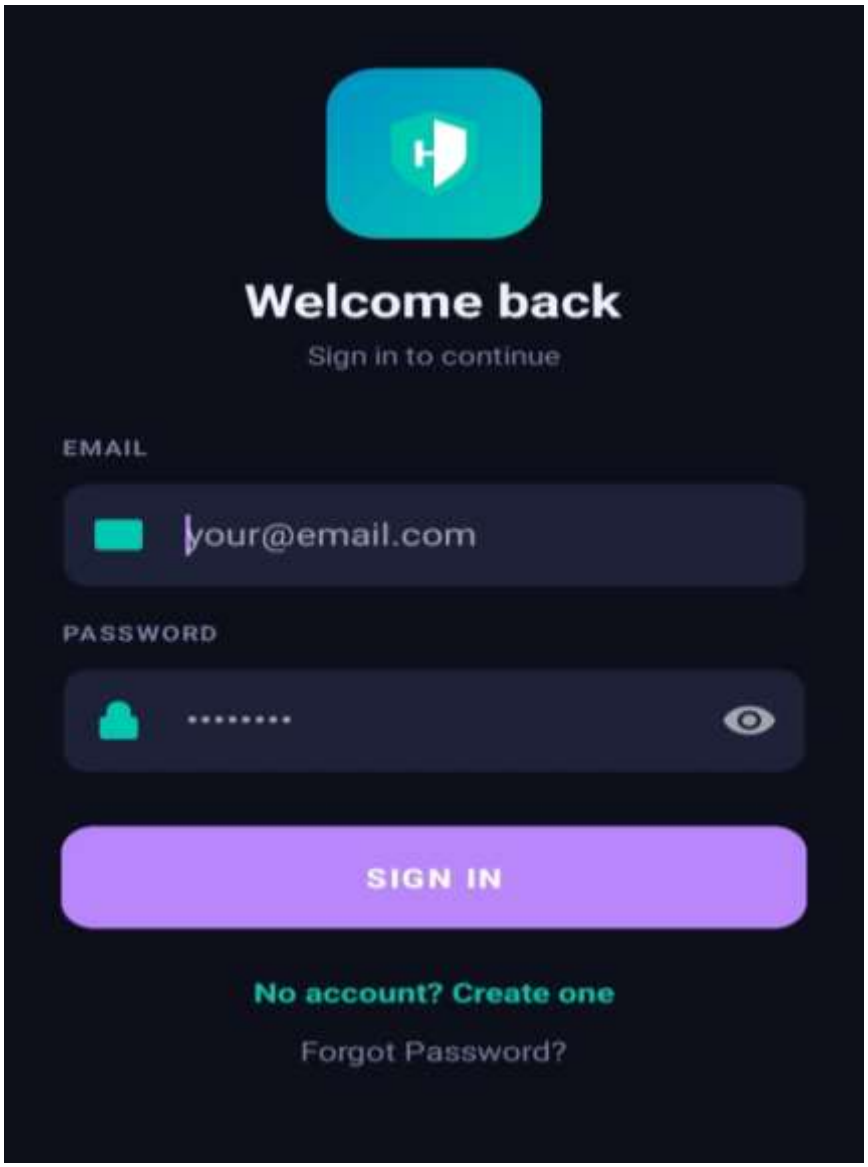

Fig 3. User login interface of the SmartReach application.

Fig 4. Passcode configuration screen of the application.

Once authentication is complete, the user is prompted to configure a secure passcode that will provide a layer of security against unauthorized access to the application. The passcode is a security layer that will prevent unauthorized access to the remote management features. As shown in Fig. 4, a numerical passcode is entered by the user that will be used for validating SMS commands.

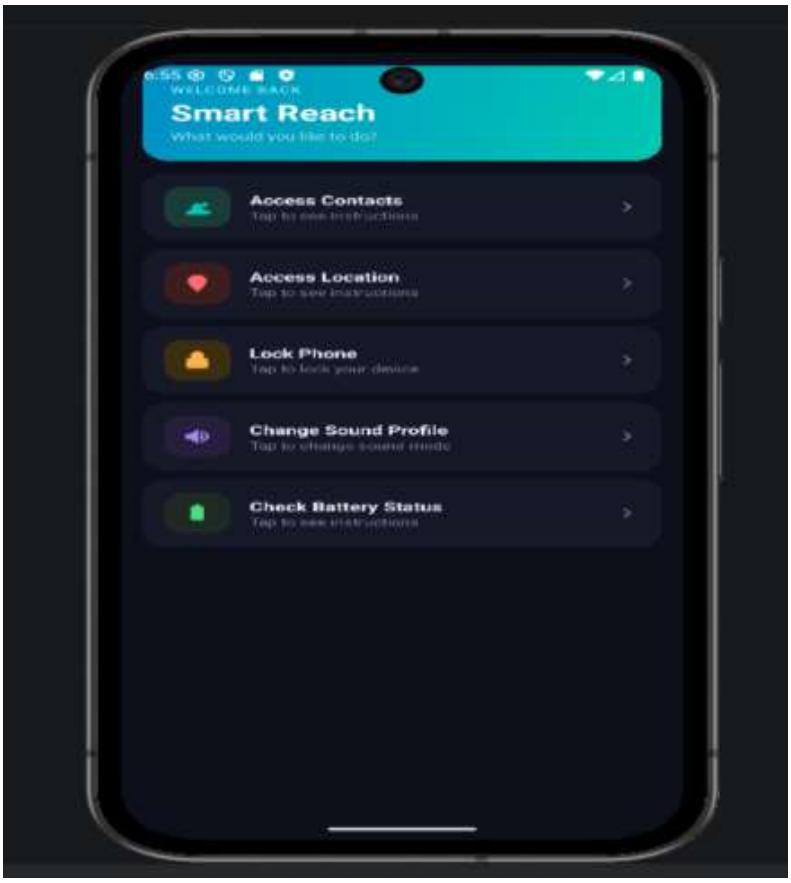

Fig 5. Dashboard of the SmartReach application displaying remote control features.

The main dashboard of the Smart Reach application gives the user access to a number of remote management facilities. As illustrated in Fig. 5, the main dashboard gives the user access to a number of facilities such as accessing the contacts, getting the location of the device, locking the phone, and changing the sound profile. These facilities allow the authorized user to remotely manage the mobile device.

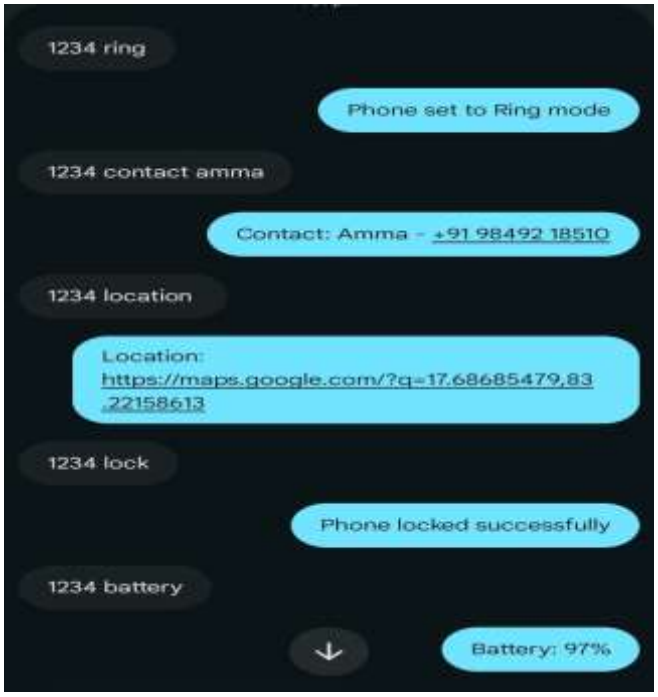

Fig.6 . SMS command execution results showing device control operations.

The system processes SMS commands from an authorized user with a predefined passcode and a command keyword. When a valid command is received, the application performs the corresponding action and sends a response message. As depicted in Fig. 7, the system is capable of processing valid commands such as locking the phone, getting the device location via a Google Maps link, and accessing stored contacts.

VII. CONCLUSION

In this paper, we developed an SMS-based Android application that enables users to remotely control their devices without requiring internet connectivity. The

application is able to perform various operations such as locking the device, retrieving GPS location, accessing contacts, and changing sound settings by utilizing Android system services BroadcastReceiver, DevicePolicyManager, and SmsManager. The project has addressed the shortcomings of existing solutions by enabling users to perform operations remotely without requiring internet connectivity. The password-protected features ensure the security of users, while GSM communication enables accessibility and response. The solution is not only practical but also beneficial to society as it offers various features like device protection, emergency management, and parental management. The project has demonstrated that system integration is a simple yet effective solution to real-world problems. The project has demonstrated that mobile applications have a great potential to provide solutions that can enhance society.

VIII. FUTURE SCOPE

However, it is found that there are areas where the system could be improved. The new versions of this application could be improved by adding more features that could provide better reliability to this system.

One such improvement that could be made is by adding encryption to the SMS commands. This will ensure that the messages are not misused by anyone else. Another improvement that could be made is by adding a remote data wipe feature. The remote data wipe feature will allow the users to wipe the data from their device in case the device is lost or stolen.

The system could be improved by adding a camera activation feature. The camera activation feature will allow the device to be enabled to capture images of the surroundings and send those images to the authorized user. The images could be used to identify the person using the device.

Other improvements that could be made to this system are by adding a SIM change alert system. The system could be improved by enabling it to handle more than one device at a time. By making such improvements to this system, it could become a more powerful mobile security system.

IX. REFERENCES

[1] A. Joy, A. P. A., A. K., and T. Devasia, "SMS-based remote mobile phone data access system," in Proc. Int. Conf. Intelligent Computing Instrumentation and Control Technologies (ICICICT), 2022.

[2] T. T. Tupe, A. A. Mhaske, A. Thube, and S. P. Shinde, "Remote android access via SMS," International Journal of Advanced Research in Science, Communication and Technology, vol. 2, no. 3, Apr. 2022.

[3] T. B. Mahadule et al., "Remote android access via SMS," IRE Journals, vol. 3, no. 9, 2020.

[4] A. Joshi et al., "Remote access of smartphone using SMS modal," TIJER, vol. 10, no. 5, May 2023.

[5] A. Jain and D. Shanbhag, "Secure SMS-based home automation system," in Proc. IEEE ICECS, 2014.

[6] R. Kumar et al., "SMS-based remote monitoring and control system for IoT sensor networks," IJIRCCE, vol. 4, no. 3, 2016.

[7] S. Khan, W. Ahmad, and R. Ali, "A research on mobile applications for location tracking through web server and SMS," IJCSIS.

[8] Android Developer, "Android permissions overview, " Google LLC.[Online]. Available:hhttps://developer.android.com/guide/topics/Permissions/overview.

[9] Android Developer, " BroadcastReceiver overview, " Google LLC.[Online]. Available:https://developer.android.com/reference/android/content/BroadcastReceiver.